IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application. No.	:	09/764,810	Confirmation No. 8729
Applicant	:	Abraham Mendelson	
Filed	:	January 16, 2001	
TC/A.U.	:	2185	
Examiner	:	Hong Chong Kim	
Docket No.	:	042390.P10140	
Customer No.	:	8791	

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

Dear Sir:

Applicant submits, the following Appeal Brief pursuant to 37 C.F.R. § 41.37 for consideration by the Board of Patent Appeals and Interferences. Applicant also submits herewith our check number 845 in the amount of $500 to cover the cost of filing the opening brief as required by 37 C.F.R. § 41.20(1)(b). Please charge any additional fees or credit any overpayment to our deposit Account No.02-2666. A duplicate copy of the Fee Transmittal is enclosed for this purpose.

10/31/2006 HDEMESS1 00000070 09764810
01 FC:1402 500.00 OP

Conclusion

Applicant respectfully requests that a timely Notice of Allowance be issued in this case.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Dated: October 27, 2006 By _____

Thinh V. Nguyen
Reg. No. 42,034
Tel.: (714) 557-3800 (Pacific Coast)

12400 Wilshire Boulevard, Seventh Floor
Los Angeles, California 90025

CERTIFICATE OF MAILING/TRANSMISSION (37 CFR 1.8A)

I hereby certify that this correspondence is, on the date shown below, being:

MAILING

☒ *deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, PO Box 1450, Alexandria, VA 22313-1450.*

Date: October 27, 2006

FACSIMILE

☐ *transmitted by facsimile to the Patent and Trademark Office.*

October 27, 2006

Tu Nguyen *Date*

TABLE OF CONTENTS

I. REAL PARTY IN INTEREST

The real party in interest is the assignee, Intel Corporation.

II. RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences known to the appellants, the appellants' legal representative, or assignee, which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

III. STATUS OF CLAIMS

Claims 1-30 of the present application are pending and claims 1-6, 10, 11-16, 20-, 21-26, and 30 remain rejected. The Applicants hereby appeals the rejection of claims 1-6, 10, 11-16, 20-, 21-26, and 30.

IV. STATUS OF AMENDMENTS

On September 7, 2005, Applicants filed a response to an Office Action dated June 7, 2005. The Examiner issued a Final Office Action on October 17, 2005. On January 9, 2006, the Applicant filed a Notice of Appeal and a Pre-Appeal Brief Review Request in response to the Final Office Action. On August 9, 2006, the Review panel issued the Notice of Panel Decision stating that the applications remains under appeal.

V. SUMMARY OF CLAIMED SUBJECT MATTER

1. Independent claims 1, 11, and 21:

A processor 105 includes an execution unit 130, a register file 150, and a Central Processing Unit (CPU) front-end 160[1]. The CPU front-end 160 includes a cache memory 140, a trace builder 110, and a trace cache subsystem 120[2].

The trace cache subsystem 120 includes, among other things, a cache manager 310, a filter trace cache (FTC) 320, and main trace cache (MTC) 330[3]. The trace cache

[1] Specification, page 3, lines 28-30; Figure 1B.
[2] Specification, page 4, lines 12-14; Figure 1C.
[3] Specification, page 5, lines 25-28; Figure 3.

subsystem 120 provides a filtering mechanism to separate and distinguish traces that are heavily used from those that do not show locality of reference. The sets are managed using a cache replacement policy such as a Least Recently Used (LRU) one[4].

The cache manager 310 manages, controls and transfers traces in the traces 101_1 to 101_N, between the different caches (e.g., FTC 320, MTC 330) in the cache subsystem 120 according to the different filtering policies and cache organizations[5]. A trace is defined as a sequence of basic blocks that have been executed consequently by the program[6].

The filter trace cache (FTC) 320 performs the filtering process. Traces 101_1 to 101_N are first inserted into the FTC 320. They stay in the FTC 320 as long as new traces do not evict them. When the traces are to be replaced, a decision is to be made on whether to discard them or move them to the MTC 330, based on their respective number of accesses[7].

When traces from traces 101_1 to 101_N are evicted from the FTC 320, the evicted traces are either discarded or kept. The kept traces are moved to the MTC 330 for long term storage. The decision to discard or promote a trace in traces 101_1 to 101_N depends on several criteria and factors. These criteria and factors may include the frequency at which traces 101_1 to 101_N are used, their lifetime within the FTC 320, and their space utilization[8]. In one embodiment, the decision to discard or to keep the trace depends on the frequency of the number of access on that particular trace[9].

2. Dependent claims 2-6, 10, 12-16, 20, 22-26, and 30:

The cache manager 310 includes, among other things, usage counters 360, a comparator 350, a management logistics 380, and storage 390. The usage counters 360_1 to 360_N are used to count the number of accesses to the traces 101_1 to 101_N [10].

The storage 390 may store a pre-determined threshold number "J". The threshold number "J" may also be dynamically adjusted. When a trace in the traces 101_1 to 101_N is

[4] Specification, page 5, lines 32-35; Figure 3.
[5] Specification, page 6, lines 1-3; Figure 3.
[6] Specification, page 5, lines 11-16.
[7] Specification, page 6, lines 21-25.
[8] Specification, page 6, lines 25-30.
[9] Specification, page 6, lines 31-32.
[10] Specification, page 6, lines 4-7; Figure 3.

to be evicted from the FTC 320, the threshold number J is used to compare to the number of accesses from a usage counter corresponding to that particularly trace[11].

When a trace is needed to be evicted from the FTC 320, the comparator 380 compares the stored threshold number J to the number of accesses extracted from the usage counters 360_1 to 360_N. If the number of accesses is equal to or higher than the threshold number "J", the trace is moved to the MTC 330. The comparator 380 compares the number of accesses of the next evicted trace from the traces 101_1 to 101_N to the threshold number "J". If it is determined that this trace has been accessed more than the threshold "J" prior to its eviction from the FTC 320, it is assumed that this trace is useful and therefore, it may be used again. This trace is then transferred to the MTC 330. If the number of accesses is less than "J", this trace is discarded[12].

In one embodiment, the threshold "J" is preset at a fixed number. In another embodiment, the threshold "J" is dynamically adjusted[13].

The trace cache subsystem 120 may further include a Level 2 (L2) trace cache 340. The L2 trace cache 340, however, may be external to the trace cache subsystem 120[14]. One of the differences between the traditional L2 cache and the L2 trace-cache 340 may be that the L2 trace-cache 340 contains traces whereas the L2 cache contains data or instructions that have not been built into a trace[15].

If a fetched instruction is a hit, the replacement information used (e.g., LRU policy) and the counter corresponds to the hit trace are updated. The trace from the first trace cache may be evicted by using a replacement mechanism[16].

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1-6, 10, 11-16, 20, 21-26, and 30 stand rejected under 35 U.S.C. §103(a) as being unpatentable over "The Cache Memory Book", Jim Handy, Academic Press, 1993, pp 37-93 ("Handy") in view of U.S. Patent No. 6,272,598 issued to Arlitt et al. ("Arlitt").

---

[11] Specification, page 6, lines 32-37.
[12] Specification, page 6, lines 37-39; page 7, lines 1-6.
[13] Specification, page 7, lines 1-2.
[14] Specification, page 7, lines 23-24; Figure 3.
[15] Specification, page 7, lines 18-20.
[16] Specification, page 8, lines 34-36; Figure 4A (Block 440); Figure 4B (Block 480)..

VII. ARGUMENTS

A. Claims 1-6, 10, 11-16, 20, 21-26, and 30 Are Not Unpatentable Over Handy In View Of Arlitt.

In the Final Office Action, the Examiner rejected claims 1-6, 10, 11-16, 20, 21-26, and 30 under 35 U.S.C. §103(a) as being unpatentable over "The Cache Memory Book", Jim Handy, Academic Press, 1993, pp 37-93 ("Handy") in view of U.S. Patent No. 6,272,598 issued to Arlitt et al. ("Arlitt"). Applicants respectfully traverse the rejection and contend that the Examiner has not met the burden of establishing a prima facie case of obviousness.

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. *MPEP §2143, p. 2100-129 (8th Ed., Rev. 2, May 2004).* Applicants respectfully contend that there is no suggestion or motivation to combine their teachings, and thus no *prima facie* case of obviousness has been established.

Handy discloses the cache memory organization. A cache unit includes a cache data RAM, a cache tag RAM, and control logic, interfacing with the CPU (Handy, Figure 2.10). A cache may be split into instruction cache to store instructions and a data cache to store data (Handy, page 60, section 2.2.3). Cache eviction is a process of writing data back to the main memory when it is being replaced in the cache (Handy, Page 63). Handy merely discloses a typical organization of a cache unit, not specifically the cache trace eviction.

Arlitt discloses a web cache performance by applying different replacement policies to the web cache. A Least Frequently Used (LFU)-aging replacement policy replaces the LFU objects, avoiding cache pollution. The LFU component maintains a frequency count for each object in the cache (Artlitt, col. 6, lines 42-55). The object is not a trace.

Handy and Arlitt, taken alone or in any combination, do not disclose, suggest, or render obvious (1) a cache manager to manage a transfer of a trace, (2) a first cache

coupled to the cache manager to evict the trace based on a replacement mechanism, and (3) a second cache coupled to the cache manager to receive the evicted trace based on a first number of accesses to the trace.

Handy merely discloses an instruction cache to store instructions, not a trace. A trace is defined as a sequence of basic blocks that have been executed by a program (See, for example, Specification, page 5, lines 10-13), not merely unorganized instructions. Furthermore, Handy merely discloses transferring an updated line to the main memory once it is to be removed from the cache (Handy, page 63, lines 24-26). A cache line is merely the smallest portion of the cache which has a unique address tag (Handy, page 64, section 2.2.5). Therefore, a line is not a trace.

Arlitt merely discloses replacement of Web objects, not a trace. Web objects are components of a Web page, such as audio files, HTML files, graphic files, or video files. They may be other files transferred via the Internet, such as File Transfer Protocol (FTP) transfers (Arlitt, col. 4, lines 30-34). Furthermore, Arlitt does not disclose a second cache to receive the evicted trace.

In the Final Office Action, the Examiner contends that Arlitt discloses a cache to receive the evicted trace based on a first number of accesses to the trace, citing Arlitt col. 6, lines 43-55 (Final Office Action, page 5, paragraph number 5, last sentence). Applicants respectfully disagree for the following reasons. To ease of reference, the cited excerpt is shown below.

> "A fifth policy is a LFU-Aging replacement policy that replaces the least frequently used object, avoiding cache pollution. The LFU component of the replacement policy maintains a frequency count for each object in the cache. The aging component of the replacement policy avoids cache pollution by reducing the frequency count of each cached object by a factor of two whenever the average frequency count exceeds a threshold parameter. Since some objects may be extremely popular, they might stay in the cache for longer than desired or have significant influence on the current frequency count. This policy retains popular objects for longer time periods and is able to discriminate against those objects which are only accessed once while in the cache (i.e., extremely unpopular objects." (Arlitt, col. 6, lines 42-55)

As seen from the above excerpt, Arlitt merely discloses a replacement policy to replace the least frequently used (LFU) object having a LFU component and an aging

component. This cache policy is not the same as the claimed invention in several aspects as discussed in the following.

First, the cached objects are Web objects, not a trace. As discussed above, a Web object is a component of a Web page. In contrast, a trace is a sequence of basic blocks that have been executed consequently by the program.

Second, the cache storage is used in a proxy server in an Internet/Intranet for caching objects received from a remote site (Arlitt, col. 3, lines 36-38). Since the objects are audio files, HTML files, graphics files, or video files, the cache storage is a mass storage subsystem such as hard drive. Therefore, it is not a cache memory used internally to the processor to store traces.

There is no motivation to combine Handy and Arlitt because neither of them addresses the problem of trace cache filtering. There is no teaching or suggestion that a trace cache is present. Handy, read as a whole, does not suggest the desirability of receiving the evicted trace based on a first number of accesses to the trace. For the above reasons, the rejection under 35 U.S.C. §103(a) is improperly made.

The Examiner failed to establish a prima facie case of obviousness and a motivation to combine the references. When applying 35 U.S.C. 103, the following tenets of patent law must be adhered to:(A) The claimed invention must be considered as a whole; (B) The references must be considered as a whole and must suggest the desirability and thus the obviousness of making the combination; (C) The references must be viewed without the benefit of impermissible hindsight vision afforded by the claimed invention; and (D) Reasonable expectation of success is the standard with which obviousness is determined. Hodosh v. Block Drug Col, Inc., 786 F.2d 1136, 1143 n.5, 229 USPQ 182, 187 n.5 (Fed. Cir. 1986). "When determining the patentability of a claimed invention which combined two known elements, 'the question is whether there is something in the prior art as a whole suggest the desirability, and thus the obviousness, of making the combination.'" In re Beattie, Lindemann Maschinenfabrik GmbH v. American Hoist & Derrick Co., 730 F.2d 1452, 1462, 221 USPQ (BNA) 481, 488 (Fed. Cir. 1984). To defeat patentability based on obviousness, the suggestion to make the new product having the claimed characteristics must come from the prior art, not from the hindsight knowledge of the invention. Interconnect Planning Corp. v. Feil, 744 F.2d 1132, 1143, 227 USPQ (BNA) 543, 551 (Fed. Cir. 1985). To prevent the use of hindsight based on the invention

to defeat patentability of the invention, this court requires the Examiner to show a motivation to combine the references that create the case of obviousness. In other words, the Examiner must show reasons that a skilled artisan, confronted with the same problems as the inventor and with no knowledge of the claimed invention, would select the prior elements from the cited prior references for combination in the manner claimed. In re Rouffet, 149 F.3d 1350 (Fed. Cir. 1996), 47 USPQ 2d (BNA) 1453. "To support the conclusion that the claimed invention is directed to obvious subject matter, either the references must expressly or implicitly suggest the claimed invention or the Examiner must present a convincing line of reasoning as to why the artisan would have found the claimed invention to have been obvious in light of the teachings of the references." Ex parte Clapp, 227 USPQ 972, 973. (Bd.Pat.App.&Inter. 1985). The mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. In re Mills, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990). Furthermore, although a prior art device "may be capable of being modified to run the way the apparatus is claimed, there must be a suggestion or motivation in the reference to do so." In re Mills 916 F.2d at 682, 16 USPQ2d at 1432; In re Fitch, 972 F.2d 1260, 23 USPQ2d 1780 (Fed. Cir. 1992).

In the present invention, the cited references do not expressly or implicitly suggest managing a transfer of a trace, evicting a trace, or receiving the evicted trace based on a first number of accesses to the trace. In addition, the Examiner failed to present a convincing line of reasoning as to why a combination of Handy and Arlitt is an obvious application of trace cache filtering.

Therefore, Applicants believe that independent claims 1, 11, 21 and their respective dependent claims are distinguishable over the cited prior art references.

VIII. CONCLUSION

Applicant respectfully requests that the Board enter a decision overturning the Examiner's rejection of all pending claims, and holding that the claims satisfy the requirements of 35 U.S.C. §103.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Dated: October 27, 2006

THINH V. NGUYEN
Reg. No. 42,034

12400 Wilshire Blvd., 7th Floor
Los Angeles, CA 90025-1026
(714) 557-3800

IX. CLAIM APPENDIX

The claims of the present application which are involved in this appeal are as follows:

1. (previously presented) An apparatus comprising:

a cache manager to manage a transfer of a trace;

a first cache coupled to the cache manager to evict the trace based on a replacement mechanism; and

a second cache coupled to the cache manager to receive the evicted trace based on a first number of accesses to the trace.

2. (previously presented) The apparatus of claim 1 wherein the cache manager comprises a usage counter to count the first number of accesses to the trace.

3. (previously presented) The apparatus of claim 1 wherein the cache manager comprises a comparator to compare a first threshold value to the first number of accesses to the trace, the first threshold value being a first fixed number or a first dynamically adjusted number.

4. (previously amended) The apparatus of claim 3 wherein the trace is transferred from the first cache to the second cache when the first threshold value is less than the first number of accesses to the trace.

5. (previously presented) The apparatus of claim 3 wherein the trace is discarded from the first cache when the first threshold value is more than or equal to the first number of accesses to the trace.

6. (previously presented) The apparatus of claim 1 further comprising a level 2 (L2) cache to receive the trace transferred from one of the first and second caches using a second number of accesses to the trace.

7. (previously presented) The apparatus of claim 6 wherein the trace is transferred from the second cache to the L2 cache when a second threshold value is less than the second number of accesses to the trace, the second threshold value being fixed or dynamically adjusted.

8. (previously presented) The apparatus of claim 6 wherein the trace is discarded from the second cache when a second threshold value is more than the second number of accesses to the trace, the second threshold value being a fixed number or a dynamically adjusted number.

9. (previously presented) The apparatus of claim 6 wherein the second number of accesses to the trace is a number of accesses to the trace counting from a time the trace first enters the first cache.

10. (original) The apparatus of claim 1 wherein the replacement mechanism is a Least Recently Used (LRU) mechanism.

11. (previously presented) A method comprising:

managing a transfer of a trace;

evicting the trace based on a replacement mechanism using a first cache; and

receiving the evicted trace based on a first number of accesses to the trace using a second cache.

12. (previously presented) The method of claim 11 wherein managing comprises counting the first number of accesses to the trace.

13. (previously presented) The method of claim 11 wherein managing comprises comparing a first threshold value to the first number of accesses to the trace, the first threshold value being a first fixed number or a first dynamically adjusted number.

14. (previously presented) The method of claim 13 wherein managing further comprises transferring the trace from the first cache to the second cache when the first threshold value is less than the first number of accesses to the trace.

15. (previously presented) The method of claim 13 wherein managing further comprises discarding the trace from the first cache when the first threshold value is more than or equal to the first number of accesses to the trace.

16. (previously presented) The method of claim 14 wherein managing further comprises transferring the trace ~~by~~ from one of the first and second caches to a second level (L2) cache using a second number of accesses to the trace.

17. (previously presented) The method of claim 16 wherein transferring comprises transferring the trace to the L2 cache when a second threshold value is less than a second number of accesses to the trace, the second threshold value being fixed or dynamically adjusted.

18. (previously presented) The method of claim 16 wherein transferring comprises discarding the trace when a second threshold value is more than a second number of accesses to the trace, the second threshold value being a fixed number or a dynamically adjusted number.

19. (previously presented) The method of claim 16 wherein the second number of accesses to the trace is a number of accesses to the trace counting from a time the trace first enters the first cache.

20. (original) The method of claim 11 wherein the replacement mechanism is a Least Recently Used (LRU) mechanism.

21. (previously presented) A system comprising:

an execution unit; and

a cache unit couple to the execution unit to provide the execution unit a trace, the cache unit comprising:

a cache manager to manage a transfer of the trace,

a first cache coupled to the cache manager to evict the trace based on a replacement mechanism, and

a second cache coupled to the cache manager to receive the evicted trace based on a first number of accesses to the trace.

22. (previously presented) The system of claim 21 wherein the cache manager comprises a usage counter to count the first number of accesses to the trace.

23. (previously presented) The system of claim 21 wherein the cache manager comprises a comparator to compare a first threshold value to the first number of accesses to the trace, the first threshold value being a first fixed number or a first dynamically adjusted number.

24. (previously amended) The system of claim 23 wherein the trace is transferred from the first cache to the second cache when the first threshold value is less than the first number of accesses to the trace.

25. (previously amended) The system of claim 23 wherein the trace is discarded from the first cache when the first threshold value is more than or equal to the first number of accesses to the trace.

26. (previously presented) The system of claim 24 wherein the cache unit further comprises a level 2 (L2) cache to receive the trace transferred from one of the first and second caches using a second number of accesses to the trace.

27. (previously presented) The system of claim 26 wherein the trace is transferred from the second cache to the L2 cache when a second threshold value is less than the second number of accesses to the trace, the second threshold value being fixed or dynamically adjusted.

28. (previously presented) The system of claim 26 wherein the trace is discarded from the second cache when a second threshold value is more than the second number of accesses to the trace, the second threshold value being a fixed number or a dynamically adjusted number.

29. (previously presented) The system of claim 26 wherein the second number of accesses to the trace is a number of accesses to the trace counting from a time the trace first enters the first cache.

30. (original) The system of claim 21 wherein the replacement mechanism is a Least Recently Used (LRU) mechanism.

XI. EVIDENCE APPENDIX

None

XII. RELATED PROCEEDINGS APPENDIX

None